Complex modulation computer-generated hologram by a fast hybrid point-source/wave-field approach

Antonin Gilles[1*] Patrick Gioia[1,2] Rémi Cozot[1,3] Luce Morin[1,4]

[1] IRT b<>com, Cesson-Sévigné, France
[2] Orange Labs, Rennes, France
[3] University of Rennes 1, Rennes, France
[4] INSA Rennes, Rennes, France

Abstract

We propose a fast Computer-Generated Hologram (CGH) computation method based on a hybrid point-source/wave-field approach. Whereas previously proposed methods tried to reduce the computational complexity of the point-source or the wave-field approaches independently, our method uses the two approaches together and therefore takes advantages from both of them. The algorithm consists of three steps. First, the 3D scene is sliced into several depth layers parallel to the hologram plane. Then, for each layer, we compute the complex wave scattered by this layer either using a wave-field or a point-source approach according to a threshold criterion on the number of points within the layer. Finally, we sum up the complex waves scattered by all the depth layers in order to obtain the final CGH. Experimental results reveal that this combination of approaches does not produce any visible artifact and outperforms both the point-source and wave-field approaches.

Keywords : Computer-Generated Hologram, Color holography, Real-time holography, Three-dimensional imaging

*This work has been achieved within the Institute of Research and Technology b<>com, dedicated to digital technologies. It has been funded by the French government through the National Research Agency (ANR) Investment referenced ANR-A0-AIRT-07. Authors can be reached at {antonin.gilles, patrick.gioia, remi.cozot, luce.morin}@b-com.com.

1 Introduction

Holography is often considered as the most promising 3D visualization technology, since it can provide the most authentic and natural three-dimensional illusion to the naked eye. Indeed, it provides complete human depth cues without the need for special viewing devices and without causing eye-strain [1]. Over the past decades, several methods have been proposed to generate holograms by computer calculation. Using these methods, it is possible to obtain Computer-Generated Holograms (CGH) of synthetic or existing scenes by simulating the propagation of light scattered by the scene towards the hologram plane. CGH computation techniques usually sample 3D scenes by a set of primitives and calculate light propagation as the sum of complex light waves scattered by each of the primitives. Commonly used primitives include points (point-source approach) and planar segments (wave-field approach).

The point-source approach samples 3D scenes by a collection of self-luminous points, and calculates complex wave scattered by each of the points using the monochromatic spherical light wave equation. This approach is very flexible and does not impose any restriction on the scene geometry. However, its complexity is very high since it requires one calculation per point of the scene per pixel of the hologram. Moreover, to produce shapes that appear solid and continuous, the scene needs to be sampled at very high densities, making the CGH computation prohibitively slow. In order to reduce the computational complexity, several methods have been proposed, including geometric symmetry [2], look-up tables [3, 4], interframe and interline redundancy reduction [5, 6], difference and recurrence formulas [7, 8], image holograms [9, 10], wave-front recording planes [11, 12, 13], using GPU hardware [14, 15], and special purpose hardware [16, 17].

The wave-field approach samples 3D scenes by a collection of self-luminous planar segments, and computes complex wave scattered by each of the segments using the angular spectrum of plane waves [18]. The computation of the angular spectrum of plane waves involves the use of the Fast Fourier Transform (FFT) algorithm twice, and is therefore more time-consuming than the computation of the spherical light wave scattered by a single point. However, complex waves scattered by scene points located within a single planar segment are calculated all at once using the angular spectrum of plane waves. Therefore, this approach is more efficient than the point-source approach when objects in a scene consist of large planar segments containing many points. However, when the scene geometry contains complex shapes, a large number of small planar segments containing only one or a few points are needed to sample it, making the wave-field approach less efficient than the point-source approach. In order to reduce the computation burden, several methods have been proposed, including the use of analytic expression of the angular spectrum [19, 20, 21, 22, 23], and color space conversion [24, 25].

In this paper, we propose a fast CGH computation method based on a hybrid point-source/wave-field approach. Whereas previously proposed methods tried to reduce the computational complexity of the point-source or the wave-field approaches independently, our method uses the two approaches together and therefore takes advantages from both of them. Section 2 gives a detailed description of our method, Section 3 gives experimental results, and Section 4 concludes this paper.

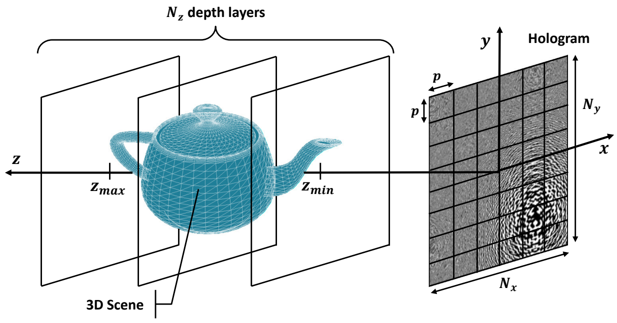

Figure 1: Scene geometry and coordinate system used by the proposed method

2 Proposed method

2.1 Overview

Figure 1 shows the scene geometry and coordinate system used by the proposed method. The coordinate system is defined by (x, y, z) so that the hologram lies on the $(x, y, 0)$ plane. The 3D scene is treated as a set of N_z depth layers parallel to the hologram plane and located between $z_{\min}$ and $z_{\max}$. The hologram is sampled on a regular 2D grid of resolution $N_x \times N_y$ with a sampling pitch p. Figure 2 shows the overall block-diagram of the proposed method, which consists of three steps. First, the 3D scene is sliced into N_z depth layers parallel to the hologram plane. Then, for each layer d, if the number of points N_d within the layer exceeds a maximum value $N_{d,\max}$ (selection criterion which will be determined in section 2.4), we compute the complex wave scattered by this layer using a wave-field approach. Otherwise, if N_d is smaller than $N_{d,\max}$, the method calculates the complex wave scattered by this layer using a point-source approach. Finally, the method sums up the complex waves scattered by all the depth layers in order to obtain the final CGH. Afterwards, the scene image can be reconstructed from the computed CGH pattern.

2.2 Wave-field approach

When the number of points N_d within layer d exceeds $N_{d,\max}$, the complex wave U_d^w scattered by this

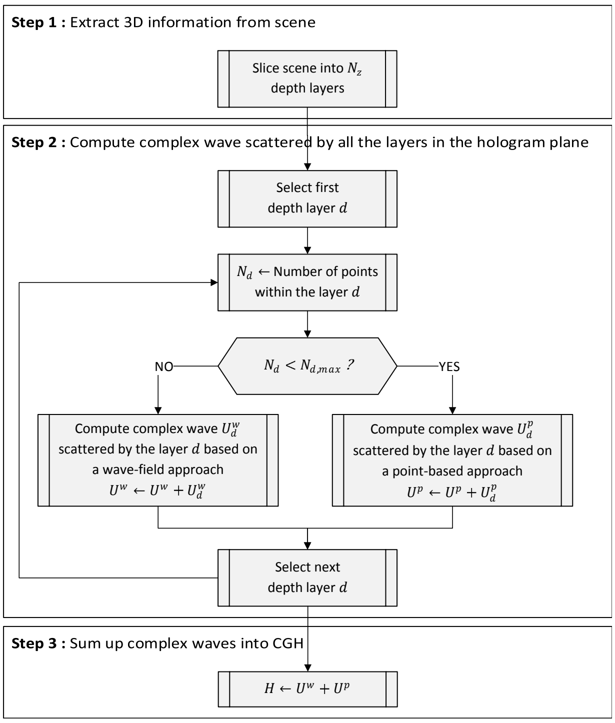

Figure 2: Block-diagram of the proposed method

layer towards the hologram plane is computed using a wave-field approach. To this end, we use the angular spectrum of plane waves [18], which is given by

$$U_d^w(x,y) = \mathcal{F}^{-1}\Big\{\mathcal{F}\left\{A_d(x,y)e^{j\phi_d(x,y)}\right\} \times e^{-j2\pi\sqrt{\lambda^{-2}-u^2-v^2}z_d}\Big\}, \quad (1)$$

where $A_d(x,y)$ and $\phi_d(x,y)$ are the amplitude and phase of the (x,y) point in layer d; λ is the wavelength of light, u and v are the spatial frequencies, z_d is the depth of layer d, and $\mathcal{F}$ and $\mathcal{F}^{-1}$ are respectively the forward and inverse Fourier Transform. These transforms can be computed using the Fast Fourier Transform algorithm (FFT). In order to render a diffusive scene, the phase $\phi_d(x,y)$ is set to a random value. Finally, the complex wave U^w scattered by all the layers whose number of points N_d exceeds $N_{d,\max}$ in the hologram plane is given by

$$U^w(x,y) = \sum_{\substack{d=0 \\ N_d > N_{d,\max}}}^{N_z} U_d^w(x,y). \quad (2)$$

In order to avoid one FFT per layer and therefore to speed-up the computation, the algorithm sums up the complex waves scattered by each layer directly in the frequency domain, and then inverse Fourier transforms the result to get U^w, as proposed in [19]:

$$\hat{U}_d^w(u,v) = \mathcal{F}\left\{A_d(x,y)e^{j\phi_d(x,y)}\right\}e^{-j2\pi\sqrt{\lambda^{-2}-u^2-v^2}z_d},$$
$$U^w(x,y) = \mathcal{F}^{-1}\left\{\sum_{\substack{d=0 \\ N_d > N_{d,\max}}}^{N_z} \hat{U}_d^w(u,v)\right\}. \quad (3)$$

2.3 Point-source approach

When the number of points within layer $d \in \{0..N_z\}$ is smaller than $N_{d,\max}$, the complex wave scattered by this layer towards the hologram plane is computed using a point-source approach. The complex wave scattered by a point source i located within layer d is given by the angular spectrum of plane waves [18] as

$$U_{d,i}^p(x,y) = A_i e^{j\phi_i}\mathcal{F}^{-1}\left\{e^{-j2\pi\sqrt{\lambda^{-2}-u^2-v^2}z_d}\right\} \otimes \delta(x-x_i, y-y_i), \quad (4)$$

where A_i and ϕ_i are the amplitude and phase of the point, x_i and y_i its coordinates within the layer, and $\otimes$ is the convolution operator. In order to avoid interference between the points, the phase ϕ_i is set to a random value.

Convolving a function with a Dirac delta shifts it around the delta impulse. Therefore, if we know the inverse Fourier transform term in Eq. (4) beforehand, $U_{d,i}^p$ can be computed simply by scaling this term with the point's amplitude and phase factor, followed by a shifting operation. In order to speed up the computation, we use a pre-calculated LUT, as proposed in [4]. The LUT $T(x,y,z)$ is pre-computed as

$$T(x,y,z) = \mathcal{F}^{-1}\left\{e^{-j2\pi\sqrt{\lambda^{-2}-u^2-v^2}z_d}\right\}h(x,y,z). \quad (5)$$

h being an envelope function used to restrict the region of contribution of a given point source, equal to one within the region of contribution of the point and zero elsewhere. This function limits the spatial frequencies of the complex wave to avoid aliasing in the CGH.

According to the Nyquist Sampling Theorem, the maximum spatial frequency $f_{\max}$ which can be represented with a sampling pitch p is given by $f_{\max} = (2p)^{-1}$. The grating equation [18] gives the relation between the maximum spatial frequency $f_{\max}$ and the maximum diffraction angle θ as $\sin(\theta) = \lambda f_{\max}$. Therefore, the region of contribution of a point source at depth z is given by its maximum radius $R_{\max}$ by

$$R_{\max} = z\tan(\theta) = z\tan\left(\arcsin\left(\frac{\lambda}{2p}\right)\right), \tag{6}$$

as shown in Figure 3. The envelope function h can thus be defined as

$$h(x,y,z) = \begin{cases} 1 & \text{if } \sqrt{x^2+y^2} < R_{\max} \\ 0 & \text{otherwise.} \end{cases} \tag{7}$$

In order to limit its number of pixels, the LUT is pre-computed only within the circumscribing square of the region of contribution defined by the envelope function h. Therefore, the number of pixels $N_{T,d}$ of the LUT for depth z_d is given by

$$\begin{aligned} N_{T,d} &= \left(\frac{2R_{\max}}{p}\right)^2 \\ N_{T,d} &= \left[\frac{2z_d}{p}\tan\left(\arcsin\left(\frac{\lambda}{2p}\right)\right)\right]^2. \end{aligned} \tag{8}$$

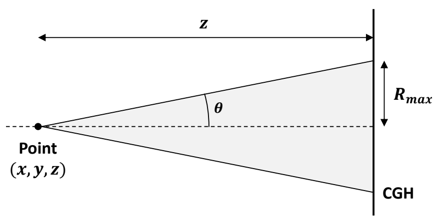

Figure 3: Region of contribution of a given point source

Then, the complex wave U_d^p scattered by layer d in the hologram plane can be obtained by simply addressing this pre-calculated LUT, as

$$U_d^p(x,y) = \sum_{i=1}^{N_d} A_i e^{j\phi_i} T(x-x_i, y-y_i, z_d). \tag{9}$$

Finally, the complex wave scattered by all the layers whose number of points N_d is smaller than $N_{d,\max}$ in the hologram plane is given by

$$U^p = \sum_{\substack{d=0 \\ N_d < N_{d,\max}}}^{N_z} U_d^p(x,y). \tag{10}$$

2.4 Determination of the selection criterion

The first step to implement the proposed method is to determine the value of $N_{d,\max}$. We call t_p the time needed to compute the complex wave scattered by a layer at depth z_d with N_d luminous points using the point-source approach presented in Section 2.3, and t_w the time needed to compute it using the wave-field approach presented in Section 2.2. Since the wave-field approach involves one complex multiplication per pixel and a Fourier transform, t_w is linearly dependent on the number of pixels of the hologram $N_{\text{pix}} = N_x \times N_y$. The point-source approach involves one complex multiplication per pixel of the LUT per point within the layer, so t_p is dependent on the number of pixels $N_{T,d}$ of the LUT for depth z_d and on the number of points N_d within the layer. t_w and t_p are experimentally found to be expressed by

$$\begin{cases} t_w(N_{\text{pix}}) = kN_{\text{pix}} \\ t_p(N_d, N_{T,d}) = \left[a\left(bN_{T,d}+c\right)^{\frac{1}{2}} + dN_{T,d} + e\right] N_d. \end{cases} \tag{11}$$

We find the numerical values for the coefficients a, b, c, d, e and k in Eq. (11) using the Gnuplot implementation of the nonlinear least-squares Levenberg-Marquardt algorithm [26]:

$$\begin{cases} a = 1,11.10^{-5} & b = 0,59 & c = 1,0 \\ d = 2,59.10^{-8} & e = 4,78.10^{-4} & k = 5,54.10^{-7} \end{cases} \tag{12}$$

In order to maximize the efficiency of our method, $N_{d,\max}$ must be set such that

$$t_p(N_{d,\max}, N_{T,d}) = t_w(N_{\text{pix}}) \tag{13}$$

$$\Leftrightarrow N_{d,\max} = \frac{kN_{\text{pix}}}{\left[a\left(bN_{T,d}+c\right)^{\frac{1}{2}} + dN_{T,d} + e\right]}. \tag{14}$$

3 Experimental results and discussion

The proposed method was implemented in C++/CUDA on a PC system employing an Intel Core i7-4930K CPU operating at 3.40 GHz, a main memory of 16 GB and an operating system of Microsoft Windows 8 as well as three GPUs NVIDIA GeForce GTX 780Ti.

For the experiments, we used the Middlebury's views and disparity maps datasets [27] as test scenes (Figures 4a and 4b). From each view and disparity map pair, a 3D point cloud is extracted, where each point is given an amplitude proportional to its corresponding pixel value in the view image and a random phase. Since each disparity map is encoded as an 8-bits gray level image, the extracted 3D point cloud is naturally sliced as a set of $N_z = 255$ depth layers parallel to the CGH plane[1]. The total number of points N_{scene} within the point cloud is given by the number of pixels of the disparity map minus the number of unknown disparity pixels. The 3D point cloud is considered to be located between $z_{\min} = -d$ and $z_{\max} = d$ in front of the CGH plane, where $2d = 2$cm is the depth extent of the scene. Finally, the CGH to be computed has a resolution of 4096×4096 with a sampling pitch $p = 8,1\mu$m.

We compare our method with GPU implementations of two other methods: (1) the wave-field method proposed in [28], which computes complex wave scattered by each layer using a wave-field approach, and (2) the point-source method proposed in [4], which computes complex wave scattered by each layer using a point-source approach. We adapted both methods to produce colorful complex modulation CGH. Figure 4 shows the scene images numerically reconstructed from the CGH patterns of the scene "Moebius" generated by the wave-field method (Figure 4d), the point-source method (Figure 4e), and our method (Figure 4f). Figure 4c shows in blue the scene points whose complex wave is computed by our method using the wave-field approach and in yellow the scene points whose complex wave is computed using the point-source approach. As seen in Figure 4, our method does not produce any visible artifact, even at the boundaries between these two categories of points.

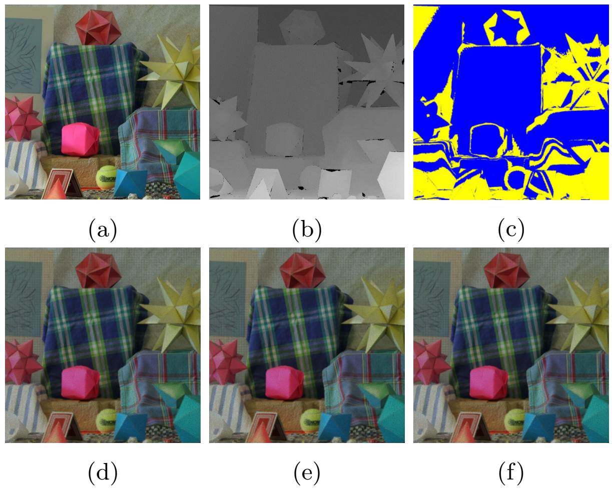

Figure 4: (a) Intensity view and (b) disparity map of the test scene "Moebius" from Middlebury's dataset. Figure (c) shows in blue the scene points whose complex waves are computed by our method using the wave-field approach and in yellow the scene points whose complex waves are computed using the point-source approach. On the second line, scene images numerically reconstructed from the CGH patterns generated by (d) the wave-field method, (e) the point-source method, and (f) our method.

In order to evaluate the objective quality of the reconstructed images compared to the original view image, we used the Peak Signal-to-Noise Ratio (PSNR). The PSNR of the reconstructed images of the scene "Moebius" were found to be $21,20$dB, $21,19$dB, and

[1]The 8-bits pixels in the disparity maps can have 256 different values, but in this dataset, the 0 value is used to encode an unknown disparity. Points with unknown disparity are not extracted from the disparity maps.

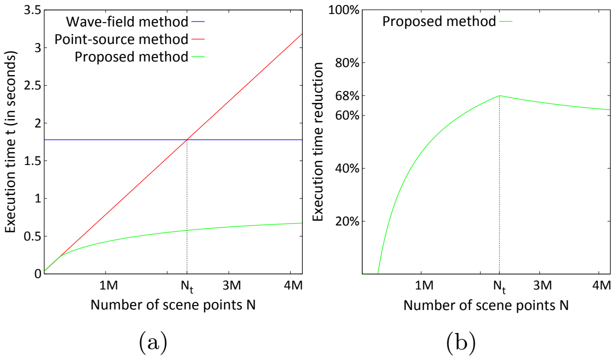

Figure 5: (a) CGH computation time for a synthetic 3D scene using the wave-field method (in blue), the point-source method (in red), and our method (in green) depending on the number of scene points N. (b) CGH computation time reduction using our method depending on the number of scene points N.

21, 20dB for the wave-field method, the point-source method and our method, respectively. These results show that our method does not reduce the quality the reconstructed scene images compared to the conventional point-source and wave-field methods. It must be noted that unlike the original view image, the numerically reconstructed images have a low depth of field due to the reconstruction technique used. As a consequence, the PSNR of the reconstructed images are found to be below 30dB. Additionally, we compared the CGH pattern generated by our method to those generated by the wave-field and point-source methods using the PSNR. The PSNR of the CGH pattern generated by our method was found to be 40, 59dB and 42, 88dB, compared to the wave-field and point-source methods, respectively.

In Figure 5a, we compared the CGH computation time of the wave-field method (in blue), the point-source method (in red), and our method (in green) depending on the number of scene points N using views and depth maps pairs of a single synthetic 3D scene with different resolutions. As shown on Figure 5a, while the computation time of the point-source method increases linearly with the number of scene points, the computation time of the wave-field method does not depend on it. Therefore, while the point-source method is faster than the wave-field method for scenes with few points, the wave-field method is still more efficient than the point-source method for scenes with a large number of points. By combining these two approaches, our method takes advantages from both of them and is therefore always the most efficient.

Figure 5b shows the reduction of the CGH computation time using our method depending on the number of scene points N. As seen in Figure 5b, our method allows the CGH computation time to be reduced by a percentage that increases quickly until N passes a threshold N_t, and then decreases slowly. This threshold corresponds to the number of scene points for which the computation time of the point-source method reaches the computation time of the wave-field method. The value of N_t depends on the number of hologram pixels $N_{\text{pix}} = N_x \times N_y$, on the number of depth layers N_z, and on the distance between the scene and the CGH plane. As shown on Figure 5b, the CGH computation time is reduced by 68% using our method when the number of scene points is equal to N_t. Moreover, our method outperforms both the point-source and wave-field methods even when the number of scene points is higher than N_t. In addition to the results shown here, we have conducted many tests on both real and synthetic scenes with different number of hologram pixels and depth layers. A reduction of over 65% of the computation time has been reached for each test scene when the number of scene points is equal to N_t. These experimental results confirm the performance superiority of our method over the conventional point-source and wave-field methods in terms of computation time.

4 Conclusion

In this paper, we proposed a fast Computer-Generated Hologram (CGH) computation method based on a hybrid point-source/wave-field approach. The algorithm consists of three steps. First, the 3D scene is sliced into several depth layers parallel to the hologram plane. Then, for each layer, if the number of points within the layer exceeds a determined maximum value, we compute the complex wave scattered by this layer using a wave-field approach. Otherwise, we compute the complex wave scattered by this layer using a point-source approach. Finally, we sum up the complex waves scattered by all the depth layers in order to obtain the final CGH. Experimental results reveal that the CGH computation time has been reduced up to 68% compared to the conventional point-source and wave-field methods without producing any visible artifact. This confirms the performance superiority of our method over the conventional point-source and wave-field methods in terms of computation time.

Our method does not take into account occlusions between objects in the scene, so in future study we plan to improve this method in order to handle scene occlusions properly.

References

[1] Ulf Schnars and Werner Jüptner. *Digital Holography: Digital Hologram Recording, Numerical Reconstruction, and Related Techniques.* Springer Science & Business Media, December 2005.

[2] J. L. Juárez-Pérez, A. Olivares-Pérez, and L. R. Berriel-Valdos. Nonredundant calculations for creating digital fresnel holograms. *Appl. Opt.*, 36(29):7437–7443, October 1997.

[3] Mark E. Lucente. Interactive computation of holograms using a look-up table. *J. Electron. Imaging*, 2(1):28–34, January 1993.

[4] Seung-Cheol Kim and Eun-Soo Kim. Effective generation of digital holograms of three-dimensional objects using a novel look-up table method. *Appl. Opt.*, 47(19):D55–D62, July 2008.

[5] Seung-Cheol Kim and Eun-Soo Kim. Fast computation of hologram patterns of a 3d object using run-length encoding and novel look-up table methods. *Appl. Opt.*, 48(6):1030–1041, February 2009.

[6] Seung-Cheol Kim, Woo-Young Choe, and Eun-Soo Kim. Accelerated computation of hologram patterns by use of interline redundancy of 3-d object images. *Opt. Eng*, 50(9):091305–091305–10, 2011.

[7] Hiroshi Yoshikawa, Susumu Iwase, and Tadashi Oneda. Fast computation of fresnel holograms employing difference. In *Practical Holography XIV and Holographic Materials VI*, volume Proc. SPIE 3956, pages 48–55, May 2000.

[8] Kyoji Matsushima and Masahiro Takai. Recurrence formulas for fast creation of synthetic three-dimensional holograms. *Appl. Opt.*, 39(35):6587–6594, December 2000.

[9] Takeshi Yamaguchi, Gen Okabe, and Hiroshi Yoshikawa. Real-time image plane full-color and full-parallax holographic video display system. *Opt. Eng*, 46(12):125801–125801–8, 2007.

[10] Hiroshi Yoshikawa, Takeshi Yamaguchi, and Ryo Kitayama. Real-time generation of full color image hologram with compact distance look-up table. In *Advances in Imaging*, OSA Technical Digest (CD), page DWC4. Optical Society of America, April 2009.

[11] Tomoyoshi Shimobaba, Nobuyuki Masuda, and Tomoyoshi Ito. Simple and fast calculation algorithm for computer-generated hologram with wavefront recording plane. *Opt. Lett.*, 34(20):3133–3135, October 2009.

[12] Tomoyoshi Shimobaba, Hirotaka Nakayama, Nobuyuki Masuda, and Tomoyoshi Ito. Rapid calculation algorithm of fresnel computer-generated-hologram using look-up table and wavefront-recording plane methods for three-dimensional display. *Opt. Express*, 18(19):19504–19509, September 2010.

[13] Jiantong Weng, Tomoyoshi Shimobaba, Naohisa Okada, Hirotaka Nakayama, Minoru Oikawa, Nobuyuki Masuda, and Tomoyoshi Ito. Generation of real-time large computer generated hologram using wavefront recording method. *Opt. Express*, 20(4):4018–4023, February 2012.

[14] Lukas Ahrenberg, Philip Benzie, Marcus Magnor, and John Watson. Computer generated holography using parallel commodity graphics hardware. *Opt Express*, 14(17):7636–7641, August 2006.

[15] Nobuyuki Masuda, Tomoyoshi Ito, Takashi Tanaka, Atsushi Shiraki, and Takashige Sugie. Computer generated holography using a graphics processing unit. *Opt. Express*, 14(2):603–608, January 2006.

[16] Tomoyoshi Ito, Nobuyuki Masuda, Kotaro Yoshimura, Atsushi Shiraki, Tomoyoshi Shimobaba, and Takashige Sugie. Special-purpose computer HORN-5 for a real-time electroholography. *Opt. Express*, 13(6):1923–1932, March 2005.

[17] Yasuyuki Ichihashi, Hirotaka Nakayama, Tomoyoshi Ito, Nobuyuki Masuda, Tomoyoshi Shimobaba, Atsushi Shiraki, and Takashige Sugie. HORN-6 special-purpose clustered computing system for electroholography. *Opt. Express*, 17(16):13895–13903, August 2009.

[18] Joseph W. Goodman. *Introduction to Fourier Optics*. Roberts and Company Publishers, Englewood, Colo, 3rd edition, 2005.

[19] Lukas Ahrenberg, Philip Benzie, Marcus Magnor, and John Watson. Computer generated holograms from three dimensional meshes using an analytic light transport model. *Appl. Opt.*, 47(10):1567–1574, April 2008.

[20] Hwi Kim, Joonku Hahn, and Byoungho Lee. Mathematical modeling of triangle-mesh-modeled three-dimensional surface objects for digital holography. *Appl. Opt.*, 47(19):D117–D127, July 2008.

[21] Hironobu Sakata and Yuji Sakamoto. Fast computation method for a fresnel hologram using three-dimensional affine transformations in real space. *Appl. Opt.*, 48(34):H212–H221, December 2009.

[22] Yuanzhi Liu, Jianwen Dong, Yiying Pu, Bingchu Chen, Hexiang He, Zilan Deng, and Hezhou Wang. A fast analytical algorithm for generating CGH of 3d scene. volume 7619, pages 76190N–76190N–8, 2010.

[23] Yuanzhi Liu, Jianwen Dong, Yiying Pu, Bingchu Chen, Hexiang He, and Hezhou Wang. High-speed full analytical holographic computations for true-life scenes. *Opt. Express*, 18(4):3345–3351, February 2010.

[24] Tomoyoshi Shimobaba, Takashi Kakue, and Tomoyoshi Ito. Acceleration of color computer-generated hologram from three-dimensional scenes with texture and depth information. volume 9117, pages 91170B–91170B–8, 2014.

[25] Tomoyoshi Shimobaba, Yuki Nagahama, Takashi Kakue, Naoki Takada, Naohisa Okada, Yutaka Endo, Ryuji Hirayama, Daisuke Hiyama, and Tomoyoshi Ito. Calculation reduction method for color digital holography and computer-generated hologram using color space conversion. *Opt. Eng*, 53(2):024108–024108, 2014.

[26] D. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the Society for Industrial and Applied Mathematics*, 11(2):431–441, June 1963.

[27] D. Scharstein and Chris Pal. Learning conditional random fields for stereo. In *IEEE Conference on Computer Vision and Pattern Recognition, 2007. CVPR '07*, pages 1–8, June 2007.

[28] Muharrem Bayraktar and Meriç Özcan. Method to calculate the far field of three-dimensional objects for computer-generated holography. *Appl. Opt.*, 49(24):4647–4654, August 2010.